<u>**REMARKS**</u>

This is in response to the final Office Action mailed 12/18/2008. Applicants have amended independent claims 33 & 39, and have cancelled independent claim 27 from further consideration in this application. Applicants are not conceding in this application that those claims are not patentable over the art of record, as the present claim amendments and cancellations are only for facilitating expeditious prosecution of the pending claims. Applicants respectfully reserve the right to pursue these and other claims in one or more continuation and/or divisional patent applications. Reconsideration of this application is respectfully requested in view of this response/amendment.

<u>**STATUS OF CLAIMS**</u>

Claims 27-28 and 30-31 are hereby canceled via the current amendment.

Claims 33-34, 36, 39 and 41 are pending.

Claims 1-26, 29, 32, 35, 37, 38 and 40 were previously cancelled.

Claims 27, 28, 30, 31, 39 and 41 stand rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter.

Claims 27, 28, 30, 31, 33, 34, 36, 39 and 41 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the article entitled, "An Efficient XML Schema Typing System," by Wang et al., hereafter "Wang" in view of the article entitled, "A Model for Compound Type Changes Encountered in Schema Evolution," by Lerner, hereafter "Lerner".

REJECTIONS UNDER 35 U.S.C. § 101

Claims 27-28, 30-31, 33-34, 36, 39, and 41 stand rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The rejection with regards to claims 27-28 and 30-31 is considered moot in light of their cancelation via the current amendment. Independent claims 33 and 39 have been amended without adding new matter, wherein the newly amended independent claims 33 and 39 are directed to statutory subject matter. Therefore, Applicants respectfully request the Examiner to withdraw the 35 U.S.C. 101 with regards to pending claims 33-34, 36, 39, and 41.

REJECTIONS UNDER 35 U.S.C. § 103(a)

Claims 27, 28, 30, 31, 33, 34, 36, 39 and 41 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the article entitled, "An Efficient XML Schema Typing System," by Wang et al., hereafter "Wang" in view of the article entitled, "A Model for Compound Type Changes Encountered in Schema Evolution," by Lerner, hereafter "Lerner". The rejection with respect to claims 27-28 and 30-31 are considered moot in view of their cancellation via the current amendment. To be properly rejected under 35 U.S.C. §103(a), the cited combination of references (i.e., Wang and Lerner) need to teach each and every feature of the rejected claims. Applicants respectfully submit that the cited combination of references (i.e., Wang and Lerner) fails to teach or suggest many of the features of the pending claims.

Wang teaches a XML typing module that validates an XML document fragment against an XML schema and annotates the validated XML document with type information. Wang

teaches a specific example of utilizing the XML typing module used as part of the runtime environment for XML-related languages that use an XML Schema as a type system.

Lerner discloses a type evolution software system that helps recognize type changes by comparing schemas to produce a transformer that can update data in a database to correspond to a newer version of the schema.

Applicants' claim 33, by stark contrast, teaches an article of manufacture comprising having computer readable program code embodied therein which implements a method for compiling a structured document schema into type annotation records, wherein the computer readable program code comprises code to: build a type hierarchy ordered tree from structured document based on a derivation of relations among types in said structured document and determining one or more tuples for each type record in said structured document; form a complete typing set of said tuples; sort said typing set by their first field; create, from sorted tuples, ambiguity typing sequences for tuples having a common first field and having a unique second field, collect and sort a third field from ambiguity typing sequences, assign a unique offset number to each sorted third field, and arranging said ambiguity typing sequences based on offset numbers; create a typing array by concatenating typing tuples in resulting ambiguity typing sequences; for each type record node, N, in created typing array, if the intersection of a set of tuples in N with any ambiguity typing sequence is not empty, then replacing first typing tuple in N by typing tuple having offset, wherein offset represents a position of an ambiguity type in a given ambiguity typing sequence; create a type indexing data structure and indicating ambiguity type for each type name; and write said created index structure to storage.

Applicants agree with the Examiner's statement on page 9 of the Office Action of 12/18/2008 that the primary reference - Wang – fails to teach the presently claimed features of: (1) creating (from sorted tuples) ambiguity typing sequences for tuples having a common first field and having a unique second field, (2) collecting and sorting a third field from ambiguity typing sequences, (3) assigning a unique offset number to each sorted third field, and (4) arranging said ambiguity typing sequences based on offset numbers. However, Applicants respectfully disagree with the Examiner that such features are taught in the Lerner reference.

Specifically, the Examiner asserts that Figures 9, 12, 18 & 27 and sections 9.2 and 11.2.2.5 of the Lerner reference teaches each of the features listed above. Figure 9 of Lerner merely shows consecutive versions of a collection of interrelated types extracted from a software testing tool. More specifically, Figure 9 illustrates schema evolution via a comparison of the old types versus the new types.

Figure 12 and section 9.2 of Lerner illustrate an automated type comparison algorithm comprising three steps: (1) a name comparison stage for comparing similar names in old and new schema types, (2) use site comparison stage for comparing schema types that use types that have already been successfully compared, and (3) an exhaustive comparison stage.

Figure 18 of Lerner merely shows an example of the derivation rules generated by Lerner's type evolution software system.

Figure 27 and section 11.2.2.5 of Lerner show an example before application of a flyweight pattern in which each Char object includes which font it uses.

However, conspicuously absent in the above-mentioned Examiner's citations of Lerner, or the entire Lerner reference, is a teaching for creating from sorted tuples *ambiguity typing sequences* for tuples *having a common first field and having a unique second field*. Also absent in the Examiner's citations of Lerner, or the entire Lerner reference, is a teaching for *collecting and sorting a third field from ambiguity typing sequences*. Further absent in the Examiner's citation of the Lerner reference is a teaching for assigning a *unique offset number* to each sorted third field and arranging the ambiguity typing sequences *based on offset numbers*.

The Examiner relies on Section 3.5.1 of Wang as teaching the feature of creating ambiguity typing sequences for tuples having a common first field. Section 3.5.1 merely teaches an improved data structure for efficient type look up. Specifically, Wang teaches the extension of the terminal dictionary by allowing the terminal dictionary to have the following four fields: *terminal*, *tok*, *ntypes*, and *type#*. The extended terminal dictionary provides the required efficient type lookup. However, it should be noted that extension of the terminal dictionary by no means anticipates or renders obvious Applicants features of creating, from sorted tuples, *ambiguity typing sequences* for tuples *having a common first field and having a unique second field*, collecting and sorting a third field from ambiguity typing sequences, assigning a *unique offset number* to each sorted third field, and arranging said ambiguity typing sequences *based on offset numbers*.

Further, on page 9 of the Office Action of 12/18/2008, the Examiner states that the mere mention of a "<u>type directory</u>" in Wang's Figure 9 teaches Applicants' feature of <u>creating a *typing array* by *concatenating typing tuples in resulting ambiguity typing sequences*</u>. Applicants respectfully disagree with the Examiner's conclusion. Although Figure 9 of Wang does show a type array, it should be emphasized that <u>Wang's type array is NOT created by concatenating type tuples in ambiguity typing sequences</u>.

Therefore, for at least the reasons set forth above, Applicants respectfully assert that the combination of Wang and Lerner fail to teach or suggest many of the features of Applicants' pending claim 33. Hence, Applicants respectfully request the Examiner to withdraw the 35 U.S.C. §103 rejection with regards to Applicants' pending claim 33, and further respectfully request allowance thereof. The above-mentioned arguments with respect to claim 33 substantially apply to independent claim 39 as it recites many similar features as independent claim 33. Therefore, Applicants respectfully request the Examiner to withdraw the 35 U.S.C. §103 rejection with regards to Applicants' pending claim 39, and further respectfully requests allowance thereof.

The above-mentioned arguments with regards to independent claims 33 and 39 substantially apply to dependent claims 34, 36, and 41 as they inherit all the features of the claim from which they depend. Therefore, Applicants respectfully request the Examiner to withdraw the 35 U.S.C. §103 rejection with regards to Applicants' pending dependent claims 34, 36, and 41, and further respectfully request allowance thereof.

SUMMARY

As has been detailed above, none of the references, cited or applied, provide for the specific claimed details of Applicants' presently claimed invention, nor renders them obvious. It is believed that this case is in condition for allowance and reconsideration thereof and early issuance is respectfully requested.

As this response has been timely filed, no request for extension of time or associated fee is required. However, the Commissioner is hereby authorized to charge any deficiencies in the fees provided or credit any over payment to Deposit Account No. 09-0460.

If it is felt that an interview would expedite prosecution of this application, please do not hesitate to contact Applicants' representative at the below number.

Respectfully submitted,

/ramraj soundararajan/

Ramraj Soundararajan
Registration No. 53,832

IP Authority, LLC.
4821A Eisenhower Ave
Alexandria, VA 22304
(703) 461-7060

March 18, 2009